International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

IJIEMR Transactions, online available on 26th Mar2021. Link
:http://www.ijiemr.org/downloads.php?vol=Volume-10&issue=ISSUE-03

DOI: 10.48047/IJIEMR/V10/I03/99

Title **Transfer learning-based Plant Disease Detection**

Volume 10, Issue 03, Pages: 469-477.

Paper Authors

G.Sumalatha, Dr Jhansi Rani Singothu, Dr S. Krishna Rao

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Transfer learning-based Plant Disease Detection

G.Sumalatha[1], Dr Jhansi Rani Singothu[2], Dr S. Krishna Rao[3]

[1]Department of Computer Science and System Engineering, Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India.

[2]Assistant Professor, Department of Computer Science and System Engineering, Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India.

[3]Professor & Head of the Department, Department of Information Technology, Sir C.R.Reddy College of Engineering, Eluru, Andhra University, Visakhapatnam, Andhra Pradesh, India.

[1]sumalathagopathoti@gmail.com,[2]dr.jhansiraniaucsse@gmail.com, [3]skrao71@gmail.com

Abstract:

Deep Neural Networks in the field of Machine Learning (ML) are broadly used for deep learning. Among many of DNN structures, the Convolutional Neural Networks (CNN) are currently the main tool used for the image analysis and classification problems. Deep neural networks have been highly successful in image classification problems. In this paper, we have shown the use of deep neural networks for plant disease detection, through image classification. This study provides a transfer learning-based solution for detecting multiple diseases in several plant varieties using simple leaf images of healthy and diseased plants taken from PlantVillage dataset. We have addressed a multi-class classification problem in which the models were trained, validated and tested using 11,333 images from 10 different classes containing 2 crop species and 8 diseases. Six different CNN architectures VGG16, InceptionV3, Xception, Resnet50, MobileNet, and DenseNet121 are compared. We found that DenseNet121 achieves best accuracy of 95.48 on test data.

1. Introduction

Today, modern technology allows us to grow crops in quantities necessary for a gentle food supply for billions of individuals. But diseases remain a significant threat to the current supply, and a outsized fraction of crops are lost every year to diseases, true is especially dire for the 500 million smallholder farmers around the globe, whose livelihoods rely on their crops doing well.

It is important to urge an accurate diagnosis of plant diseases for global health and wellbeing. Disease detection in plants plays a significant role in agriculture, they have to be detected and diagnosed as early as possible, otherwise plants will be effected to serious problemswith respective to product quality and quantity.. Plant diseases cause a periodic outbreak of diseases leading to large-scale death which severely affects the economy. These problems have to be solved at the initial stage, to save the lives and money of individuals. It is very important to classify plant diseases and diagnose the plant diseases at early stages so that appropriate and timely action will be taken by the farmers to avoid further losses.

The system has to be automated, help identify plant diseases by the plant's appearance and visual symptoms might be of great help. These are often deployed in agricultural fields so as that the whole pipeline may be automated. This would not only lead to better efficiency as machines could perform better than humans in these redundant tasks but also improve the productivity of the farm. Our work solves the above-mentioned problem of automating disease classification using deep learning and computer vision techniques.

The aim of the present work is to introduce Deep Learning as transfer learning based convolutional neural networks for classifying plant diseases, focusing on images of leaves. This paper analyses the performance of six different state of the art CNN Models [5-10].

Transfer learning may be a machine learning method that focuses on storing knowledge gained while solving one problem and reuse it because the start line for a special but related problem. Transfer learning is categorized into three sub-settings: inductive transfer learning, transductive transfer learning, and unsupervised transfer learning, based on different situations between the source and target domains and tasks. In inductive transfer learning, the source and target tasks are different irrespective of source and target domains. In this case, some labelled data in the target domain are required to induce an objective predictive model in the target domain to achieve high performance in the target task by transferring knowledge from the source task.

2. Related Work

A lot of research has been exhausted the last decade on disease detection using deep learning and computer vision. Machine Learning approaches like Support Vector Machines (SVM), using K-Nearest Neighbours (KNN), K-means and Artificial Neural Networks (ANN) are used for plant disease detection.DeepLearning based disease classification models includes the

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

utilization of a spread of CNN models like AlexNet, GoogleNet, VGGNet etc. It is seen oftentimes as the dataset size is not enough, multi class classification with a lot of classes requires careful hyperparameter tuning to avoid over fitting as the model could easily get stuck in a local minimum.

(Angie K. Reyes, 2014) used pre-trained convolutional neural network using 1.8 million images from ImageNet dataset [1] and used a fine-tuning strategy to transfer learned recognition capabilities from general domains to the particular challenge of plant identification task [2]. Then they proceed to fine-tune the network for the plant identification task. Fine-tuning a network may is also a procedure supported the concept of transfer learning. Initially they begin training a CNN to learn features for a broad domain with a classification function targeted at minimizing error in that domain. Then, they replaced the classification function and optimize the network again to attenuate error in another, more specific domain. Under this setting, they were transferred features and therefore the parametersof the network from the broad domain to the specific one. This is often useful to take advantage of big visual data available on the web, and then transfer general recognition abilities to specific domains.

(Shreya Ghosal, 2020) used transfer Learning for developing deep learning model. Their proposed CNN architecture is predicated on VGG-16 and is trained and tested on the dataset collected from rice fields and therefore sthe internet. The accuracy of the proposed model is 92.46% [3].

(Cristian Iorga, 2019) ―They presented a model of Deep Convolutional Neural Networks (CNN) supported transfer learning for image recognition. The results of the pre-training phase are transferred to the problem of classification for the images belonging to the UC Merced Land Use dataset with 21 classes. As benchmark, they need considered a Deep CNN trained with a fraction of the identical UC Merced Land Use dataset containing the test images for classification. Their experimental results have shown the obvious advantage of the Deep CNN with transfer learning (accuracy of 0.87 using pre-training over 0.46 for fully training on the same dataset).

3. Proposed Method

3.1. Deep Convolutional Neural Networks

A Convolutional Neural Network (CNN) could be a stack of non-linear transformation functions that are learned from data. CNNs were originally proposed within the 1980's for digit recognition, and are recently revisited for large scale image recognition problems. The success of recent CNNs relies on several factors that include: availability of huge datasets, more computing power and new ideas and algorithms. Among the foremost successful ideas that make CNNs a robust tool for image recognition nowadays is that he concept of deep architectures [11].

A deep CNN consist of multiple layers that incrementally compute features from input images. The deep architecture proposed by Krishevsky et al. [8] demonstrated the facility of deep CNNs for the first time during a large-scale image classification setting. Deep CNNNs mainly consist of two types of layers: convolutional layers and fully connected layers. Convolutional layers could ever be understood as banks of filters that transform an input image into another image, highlighting specific patterns. On the opposite hand, fully connected layers take a vector as input and produce another vector as output. A further prediction layer is added to the highest of the network to get classification.

3.2. Transfer Learning

Transfer learning shown in figure 1 focuses on improving the learning of predictive function in target domain using knowledge in both source and target domains when domains or tasks between source and target differ. In transfer learning, layers with reusable features of a pre-trained model trained on a large readily available dataset and on a completely different task are determined. The features obtained from that layer is used as input to train a much smaller network that requires smaller number of parameters. This smaller network only needs to learn the relations for the specific problem having already learnt about patterns in the data from the pre-trained model. Transfer learning virtually creates a shallow network within a deep network by utilizing previously learned knowledge.

Transfer learning is employed when training dataset has a smaller amount of data and is similar to the pre-trained dataset. Transfer learning is implemented either by creating a suitable model from scratch, training the model on dataset with plethora of akin data or by reusing state-of-the-art model pre-trained on standard dataset with surfeit and analogous data. The latter is preferred as this overcomes the hassle of creating a model and saves time for training on different sets of data. While classifying diseased plant leaf images based on ImageNet dataset [3], the source task T_S and the target task T_T are different (i.e. $T_S \neq T_T$) because the label spaces between these two tasks are different (i.e.$Y_S \neq Y_T$). Inductive transfer learning is the best solution to solve such problem.

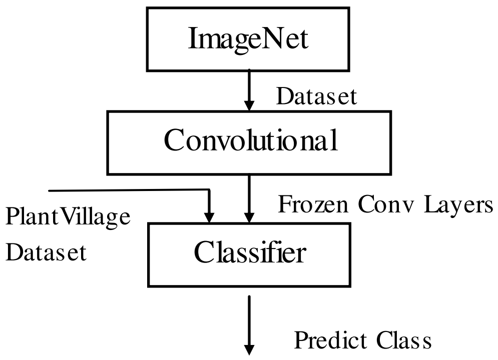

Figure 1: TransferLearning

Freezing a layer in transfer learning is referred to as not updating layer weights during training. This is done to avoid making changes on previously extracted better features by filters in the earlier layers. New dataset is trained over unfrozen layers or trainable layers. Parameters corresponding to the frozen layers remain as non-trainable parameters, whereas the network trains with remaining trainable parameters. In contrast to the back propagating and updating the weights of all the layers in the network, this is a huge decrease in computation time. The greater number of layers are frozen, the fewer the number of trainable parameters and hence lesser the computational time.

Pre-trained models can be used as a feature extractor by transferring reusable features. All the blocks except the final classifier is frozen. The number of trainable block while using the model as feature extractor is zero leading to least possible number of trainable parameters in the model. Since the models were pre-trained on ImageNet dataset, the features reused were completely obtained from ImageNet. ImageNet is an image database with more than 14 million images from more than 20000 classes. State-of-art models like VGG [5], InceptionV3 [6], Xception[7], MobileNet [8], ResNet50 [9], and Densenet121[10]pre-trained on ImageNet, were trained on new smaller datasets.

4. Dataset

This dataset is a subset of PlantVillage dataset [13] that contains plant leaf images collected from 15 different plants. We have used here 10 classes of potato and tomato healthy and infected leaves. 8984 images have been used for training, 1176 images have been used for validation and 1173 images have been used for testing. The Figure 2 shows the sample images of 10 different classes considered for classification.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Figure 2: a-f shows Potato and Tomato healthy and infected leaves a) Potato Early blight b) Potato healthy c) Potato Late Blight d) Tomato Target Spot e) Tomato Mosaic Virus f) Tomato Bacterial Spot g) Tomato Early Blight h) Tomato Healthy i) Tomato Septoria leaf j) Tomato Leaf Mold.

5. Experimental Setup

5.1Pre-trained CNN model as a Feature Extractor

To begin the transfer learning task we need to pertain the model. A pre-trained model is that the basic ingredient required starting with the task of transfer learning. A pre-trained model is an already pre-trained on a huge dataset (ImageNet) , having learned a good representation of features for over a million images belonging to 1,000 different categories, can act as a good feature extractor for new images suitable for computer vision problems. The images that need to be classified may never exist in the ImageNet dataset or may be of totally different categories, but the model should still be able to extract relevant features from these images.

In this paper, we are utilizing a pre-trained model as a feature extractor. Thedeep learning model is essentially a stacking of interconnected layers of neurons, with the final layer acting as a classifier. This architecture enables deep neural networks to capture different features at different levels in the network. We can make use this property to extract features. This is made possible by removing the final layer and freezes the convolutional base. The output of convolutional base given as input to the classification layer. For extracting the features, we have been used six different CNN models VGG16, InceptionV3, Xception, Resnet50, MobileNet, and DenseNet121.

5.2 Classification

We build the model with one fully connected layer of 256 neurons and dropout layer [15] at rate of 0.5 and finally classification layer with 10 neurons for predicting 10 different classes. The model has been compiled with two different optimizers RMSprop and Adam with various learning rates. We train the model using the extracted train and validation features for 30 epochs and haveset the batch size to 30.

Table 1 and 2 shows the accuracies obtained while employing transfer learning on

CNN Model	RMSprop Optimizer		
	$\alpha = 1e^{-5}$	$\alpha = 2e^{-5}$	$\alpha = 5e^{-5}$
VGG16	86.61	90.02	91.90
InceptionV3	86.87	87.80	87.89
Xception	91.30	92.15	92.75

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

MobileNet	93.77	95.65	95.99
Resnet50	59.67	64.62	68.20
DenseNet121	93.94	95.48	93.94

Table 1: Accuracies of various state-of-the-art CNN models with RMSprop Optimizer

CNN Model	Adam Optimizer		
	$\alpha = 1e^{-5}$	$\alpha = 2e^{-5}$	$\alpha = 5e^{-5}$
VGG16	92.49	90.62	92.24
InceptionV3	86.78	87.72	88.57
Xception	92.75	90.96	92.49
MobileNet	94.28	94.62	94.88
Resnet50	60.18	65.98	67.00
DenseNet121	93.60	95.39	95.05

Table 1: Accuracies of various state-of-the-art CNN models with Adam Optimizer

PlantVillage dataset on six state-of-the-art CNN models with various learning rates and optimizers.

In the given Table 1 and 2 ,we found that DensNet121 attained best accuracy of 95.48 with RMSprop optimizer at a learning rate 0.00002. The corresponding accuracy and loss plots are shown in Figure 3.

The graph in figure 4 shows us the performance of DenseNet121 model at various learning rates with RMSprop optimizer and Adam optimizer.

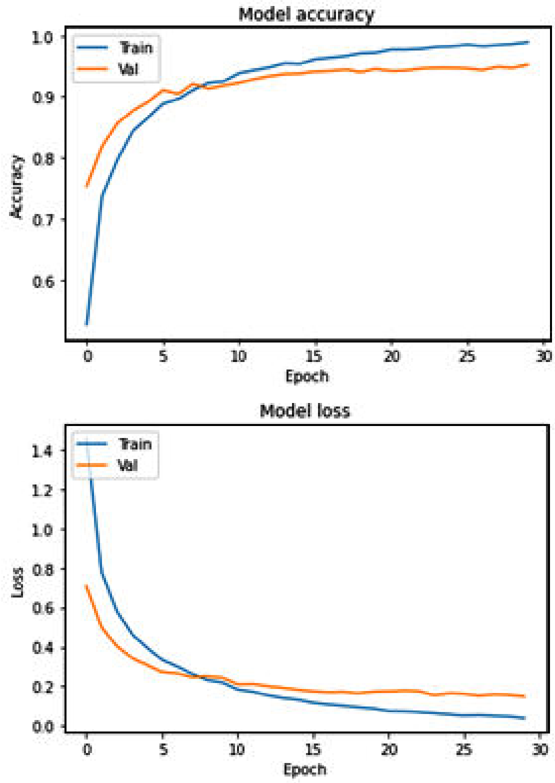

Figure:3 Accuracy and loss plots of DenseNet121 Model with RMSprop optimizer at a learning rate 0.00002.

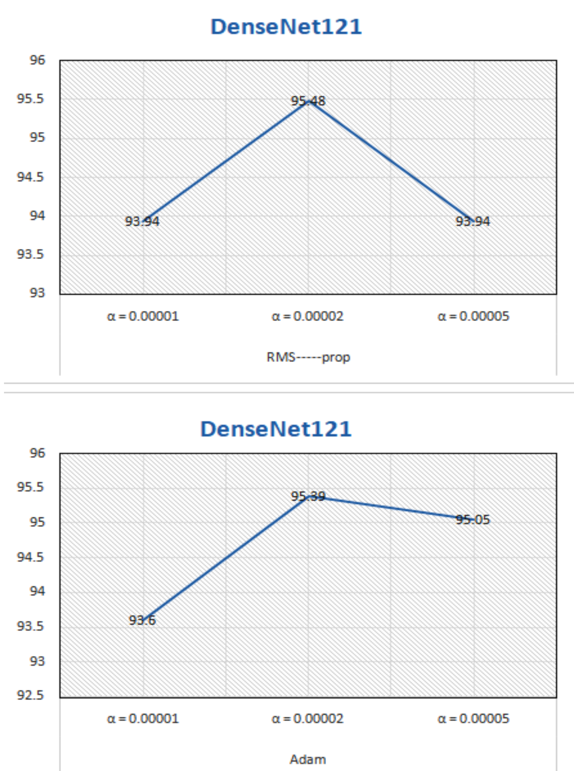

Figure 4: Performance of Densenet121 architecture.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

5 Conclusions

Diseases in plants reduce the crop yield that in turn creates threat to food supply. Hence early detection of diseases causes quick recovery of plants. This paper demonstrates the automatic disease detection with deep learning using convolutional neural network approach through image classification. We have used publicly available PlantVillage dataset of 11,333 images of diseased and healthy plant leaves, a deep convolutional neural network is trained to classify 10 different classes containing 2 crop species and 8diseases. In this paper, a transfer learning method was adapted to get general features from large ImageNet dataset, later classification applied on more specific dataset in order to automatically classify and detect plant diseases from leaf images. The complete procedure was described, respectively, from collecting the images used for training and validation and finally the procedure of training the deep CNN. We summarized the final results and came to the conclusion that DensNet121 achieved the highest accuracy of 95.48 on test data.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei, "ImageNet: A Large-Scale Hierarchical Image Database," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[2] A. K. Reyes, J. C. Caicedo, and J. E. Camargo. Fine-tuning deep convolutional networks for plant recognition. CLEF (Working Notes), 1391:467–475, 2015.

[3] Shreya Ghosal , Kamal Sarkar: "Rice Leaf Diseases Classification Using CNN With Transfer Learning", Proceedings of 2020 IEEE Calcutta Conference (CALCON).

[4]Cristian Iorga, Victor-Emil Neagoe: "A Deep CNN Approach with Transfer Learning for Image Recognition", ECAI 2019 - International Conference – 11th Edition, Electronics, Computers and Artificial Intelligence.

[5] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," CoRR, vol. abs/1409.1556, 2014.

[6] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 2818–2826.

[7] Franc¸ois Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions", Computer Vision and Pattern Recognition (cs.CV), 4th Aprial 2017.

[8[Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko and co. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", Computer Vision and Pattern Recognition (cs.CV), 17th Aprial 2017.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 770–778.

[10] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017, pp. 2261–2269.

[11]Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012).

[12] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In Advances in neural information processing systems, pages 3320–3328, 2014.

[13] David P. Hughes and Marcel Salathe, "An open ac- ´ cess repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing," arXiv preprint arXiv:1511.08060, pp. 1–13, 2015.)

[14]. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. arXiv preprint arXiv:1409.4842 (2014)

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1): 1929–1958, 2014.